LAWRENCE
LIVERMORE
NATIONAL
LABORATORY

UCRL-CONF-206883

Automatic Identification of the Templates in Matched Filtering

A. A. S. Awwal

September 29, 2004

SPIE Annual Meeting 2004
Denver, CO, United States
August 2, 2004 through August 5, 2004

Disclaimer

This document was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor the University of California nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or the University of California. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or the University of California, and shall not be used for advertising or product endorsement purposes.

Automatic identification of the templates in matched filtering

Abdul Ahad S. Awwal
National Ignition Facility
Lawrence Livermore National Laboratory, Livermore, CA. 94551
E-mail: awwal1@llnl.gov

ABSTRACT

In laser beam position determination, various shapes of markers may be used to identify different beams. When matched filtering is used for identifying the markers, one is faced with the challenge of determining the appropriate filter to use in the presence of distortions and marker size variability. If the incorrect filter is used, it will result in significant position uncertainty. Thus in the very first step of position detection one has to come up with an automated process to select the right template to use. The automated template identification method proposed here is based on a two-step approach. In the first step an approximate type of the object is determined. Then the filter is chosen based on the best size of the specific type. After the appropriate filter is chosen, the correlation peak position is used to identify the beam position. Real world examples of the application of this technique from the National Ignition Facility (NIF) at Lawrence Livermore National Laboratory are presented.

Key word: pattern recognition, matched filtering, optical alignment, automated optical alignment, automated target recognition.

1. INTRODUCTION

The National Ignition Facility, currently under construction at the Lawrence Livermore National Laboratory, is a stadium-sized facility containing a 192-beam, 1.8-megajoule, 500-terawatt, ultraviolet laser system for the study of inertial confinement fusion and the physics of matter at extreme temperatures and pressures [1]. Automatic alignment based on video images allows one to align the laser beams quickly and accurately enough to meet system requirements. At the heart of this technique is the beam position detection algorithm, which determines the position of beam features of images taken along the beam path. Varieties of alignment markers may be utilized to designate various beams, such as reference beams, main beams etc. In situations where multiple beams may need to be co-aligned, use of different markers can facilitate unique identification of each of the beams. In such situations, it is necessary to identify the position of each of the markers separately. Here we present a method that will automatically choose the template that best matches the desired pattern and use it to determine the position. Thus this is a multiclass, multiple target detection problem. Applications of this technique will also be appropriate in automated target recognition, where one has to ascertain the reliability of using a certain matched filter.

2. BACKGROUND

The *classical matched filter* (CMF) [2] and its variation *phase only filter* (POF) [3] are popular methods for matched filtering. In the CMF, the complex amplitude and phase of the reference pattern is used, whereas POF uses only the phase of the reference pattern to perform correlation [3]. The *amplitude modulated phase only filter* (AMPOF) was designed to further enhance the performance of the POF [4] by modulating the POF by an inverse type of amplitude.

The mathematical foundation of the matched filter is derived as follows. Let the Fourier transform of the object function $f(x, y)$ be denoted by:

$$F(U_x, U_y) = \left|F(U_x, U_y)\right| \exp(j\Phi(U_x, U_y)) \quad (1)$$

A Complex Match Filter (CMF) corresponding to this function f (x, y) is expected to produce its autocorrelation. From the Fourier transform theory of correlation, the CMF is given by the complex conjugate of the input Fourier spectrum as denoted by Eq. 2.

$$H_{CMF}(U_x, U_y) = F^*(U_x, U_y) = \left|F(U_x, U_y)\right| \exp(-j\Phi(U_x, U_y)) \quad (2)$$

The inverse Fourier transformation of the product of F (U_x, U_y) and H_{CMF} (U_x, U_y) results in the convolution of f (x, y) and f (-x, -y), which is the equivalent of the autocorrelation of f (x, y). Moreover, when $|F$ (U_x, U_y)| is set to unity, H_{CMF} becomes a *phase only filter* (POF):

$$H_{POF}(U_x, U_y) = \exp(-j\Phi(U_x, U_y)) \quad (3)$$

The phase only correlation of input image and the target is simply:

$$C_{POF}(\Delta x, \Delta y) = F^{-1}\left\{F\left(U_x, U_y\right) H_{POF}\left(U_x, U_y\right)\right\} \quad (4)$$

A more generalized treatment [5] of the AMPOF can be given as follows, where the generalized AMPOF filter is expressed as:

$$H_{AMPOF}(U_x, U_y) = \frac{aF^*(U_x, U_y)}{\left[\, b + c\left|F(U_x, U_y)\right| \,\right]^m} \quad (5)$$

When, a = b, m = 1, c = 0, this results in the classical matched filter; when b = 0, a = c and m = 1 it becomes a phase only filter expressed by Eq. 3. When b is a small constant for nonzero values of a, and c and m $\geq$ 1 the result is an AMPOF [3]. The position of the object can be found from the position of the cross-correlation, auto-correlation, and the position of the template using Eqs. 6-7.

$$x_{pos} = x_{cross} - x_{auto} + x_c \quad (6)$$

$$y_{pos} = y_{cross} - y_{auto} + y_c \quad (7)$$

Where ($x_{pos,}$ y_{pos}) is the to-be-determined position of the pattern in the image plane, ($x_{auto,}$ y_{auto}) is the position of the template autocorrelation peaks and (x_{cross}, y_{cross}) is the position of the crosscorrelation peak. The position of the cross-correlation peak was estimated using a polynomial fit to the correlation peak. The center of the template, ($x_{c,}$ y_c), and ($x_{auto,}$ y_{auto}) may be calculated off-line.

3. ALGORITHM DESCRIPTION

3.1 Simple example: fixed size template

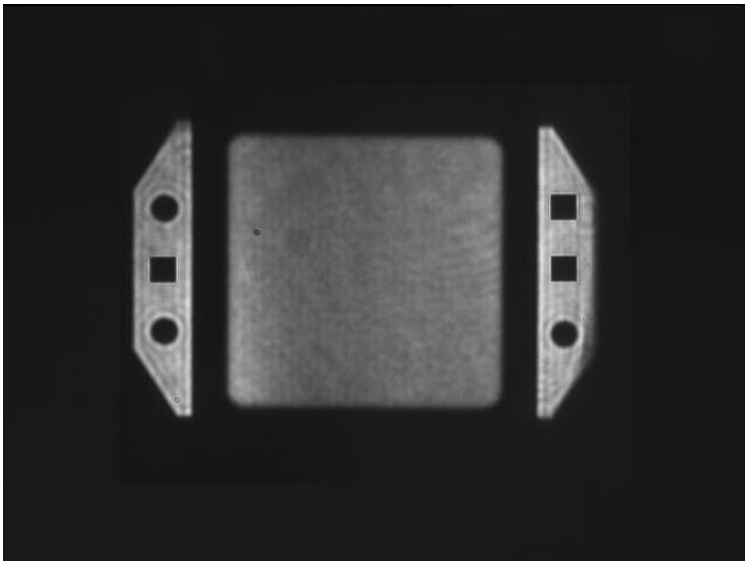

Fig. 1. Two classes of patterns

As an example, consider the image shown in Figure 1. It contains two types of objects: circles and squares. The diameter of the circle is equal to the side of the square. If we use a single filter corresponding to a circle to locate the position of the objects in the beam, then the result is shown in Fig. 2. Note that this filter produces correlation peaks for both types of objects. Thus it is possible to use only a single matched filter corresponding to the circle to detect the position of both objects.

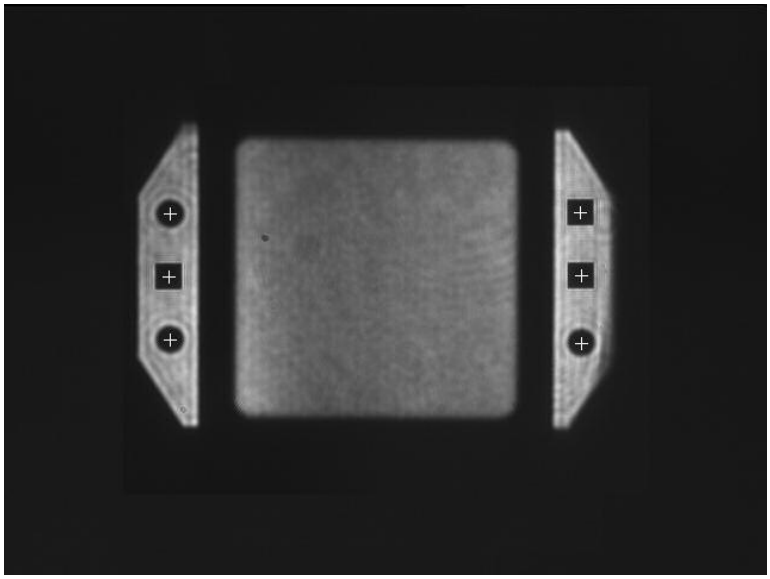

Fig. 2. Two classes of patterns with positions identified

Next we attempt to increase the detection accuracy by increasing the discrimination between the two to-be-detected objects. Preprocessing the image and getting the features of the objects can achieve this. Thus we perform edge detection on the object as shown in Fig. 3. Now instead of using circles we use the edge of the circles as filters and the resulting correlation is shown in Fig. 4. Note however, that the autocorrelation of the circle is higher than the crosscorrelation with the squares. Now based on the normalized autocorrelation value, we can set a dynamic threshold (as a percentage of the maximum peak) to reject the correlation due to non-circles. The correlation plane output depicted in Fig. 4 demonstrates a point made earlier in the previous example that both the auto and cross-correlation can identify the two beam features, if the threshold is lowered. After selecting the circles, we correlate the image from Fig. 3 with a second template consisting of a square mask. In this case, we have chosen the appropriate filter for the to-be-recognized object. Now using Eqs. 6-7, the position of the objects can be found from the position of the crosscorrelation peak, the autocorrelation peak, and the template.

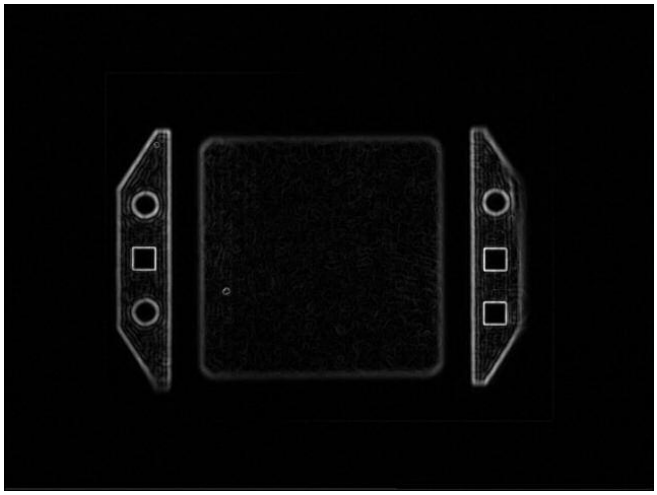

Fig. 3. The edge of the image in Fig. 1

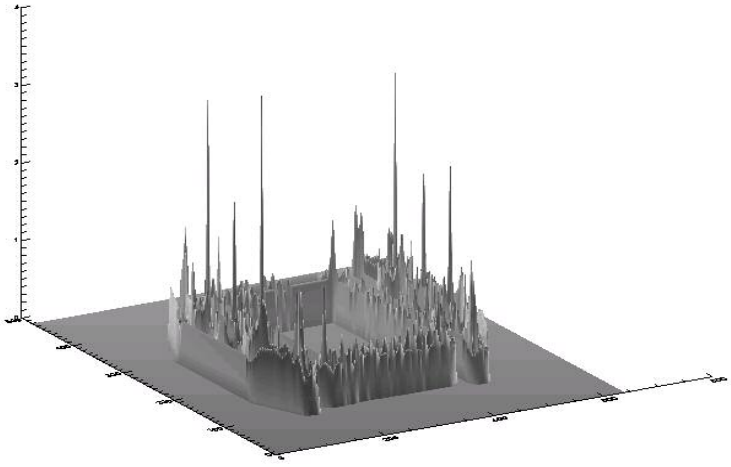

Fig. 4. The correlation with circle of the image in Fig. 3

3.2 A more complex problem: Variable dimension of templates

In the last example, we used our knowledge of the exact size of beam features from design specification. Now assume that the circles and squares could vary in size due to noise and other varying imaging conditions. Assume also that the radii of the circles could vary between range $r_1 < r < r_2$ and the sides of the squares, d could vary between d_1 and d_2. This

information can be exploited to isolate each object based on the pixel counts of each connected region. Also we assume that sufficient smoothing has been done to ensure that each connected region remains connected.

Thus by searching each connected region based on its size between πr_1^2 and $\pi r2^{2,}$ we can separate each region of interest to include only one of the patterns. A search in this region with the radius of the circle varying between $r_1 - \Delta < r < r_2 + \Delta$ will provide the correlation peaks for each r value (as shown in Table 1 below) from which we can find the best radius of the circle category. Similarly, we can find the best radius for a different sized circle. Next we can isolate the regions of each square and run a search algorithm by changing the side of the square.

Once all the parameters for each of the objects are selected we can then search the whole image with each template to find the occurrence of that type of marker. Again, using a threshold criterion based on the maximum normalized autocorrelation we can reject other objects. A typical run for a search is shown in the table below:

Table 1. A typical output of a search routine

```
****************Beginning a parameter search.....................
Found           6 interesting spots with areas in the range
      102.400    to   491.520 pixels
         422         425         406         403         405         393
Looking for square located near      x=142.000      y=228.000
Looking for big circles located near x=138.500      y=176.000
Searching for the squares
Max corr is      1.12061 at radius      8.50000
Max corr is      1.61701 at radius      9.00000
Max corr is      2.13849 at radius      9.50000
Max corr is      1.77429 at radius      10.0000
Max corr is      1.22248 at radius      10.5000
Max corr is      1.00692 at radius      11.0000
Searching for the big circles
Max corr is      1.05390 at radius      10.5000
Max corr is      1.33643 at radius      11.0000
Max corr is      1.62407 at radius      11.5000
Max corr is      1.49414 at radius      12.0000
Max corr is      1.11687 at radius      12.5000
*******Search of parameter (circle and square dimensions) completed....
Chosen Circle radius = 11.5000 Chosen Square side = 9.50000
****************Finding the positions.................................
Looking for the squares with sq_side      9.50000
xpos=       138.043 ypos=     228.941 corr=     0.0285132 spots=       1
xpos=       483.978 ypos=     228.078 corr=     0.0282019 spots=       2
xpos=       483.451 ypos=     175.233 corr=     0.0241553 spots=       3
Looking for the circles with radius      11.5000
xpos=       139.452 ypos=     176.383 corr=     0.0216542 spots=       4
xpos=       139.148 ypos=     282.037 corr=     0.0190557 spots=       5
xpos=       484.264 ypos=     284.549 corr=     0.0182978 spots=       6
Possible       6  positions found...........
```

Note that the same steps are repeated twice for the squares and circles. Assuming there are four types of objects (two objects, each with two different sizes), each of the above steps will be carried out four times to identify the location of all the objects.

One of the characteristics of the previous algorithm is that the exact radius is not known, but the range is known. In another application, where we are expecting a range of feature size because of defocus or different sized pinholes, we might need automated range identification. In these cases noise complicates the situation. In this case, we may know that we are looking for a circular spot, but because the range could be very large, we also need to guess the range $r_1 < r < r_{2.}$ In other words, the range $[r_1, r_2]$ is unknown. In some cases the beam radius may vary from 5 to 40, some other cases

from 35 to 250 pixels as shown in Fig. 5. In order to reduce the processing time, instead of searching the whole range from 35 to 200, we would prefer to estimate the range to a smaller interval. In these cases, we may want to take a different approach.

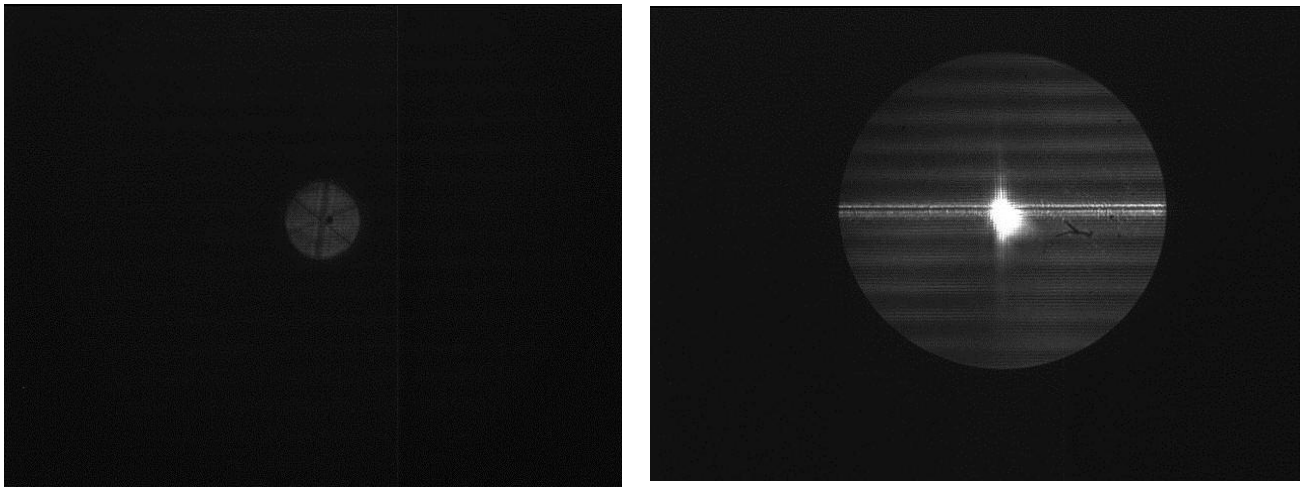

Fig. 5. Images with size variations from 35 to 200 pixels radius

Our approach to dealing with these wide variations is to add another step of estimation of the approximate radius, for example, performing a region of interest, followed by an estimation process. In the estimation, multiple techniques have yielded success for various sizes of beams. For example, in one case smoothing followed by counting the number of pixels that are within 70% of the maximum amplitude of the illuminated area gives us an approximate idea of the feature size. In other cases, we may perform a successive approximation technique where a line segment through the pattern is used to get an approximate radius in various directions. Then using this as a starting point, a search is carried out through a range of radii. In some instances voting is performed to find the most likely range.

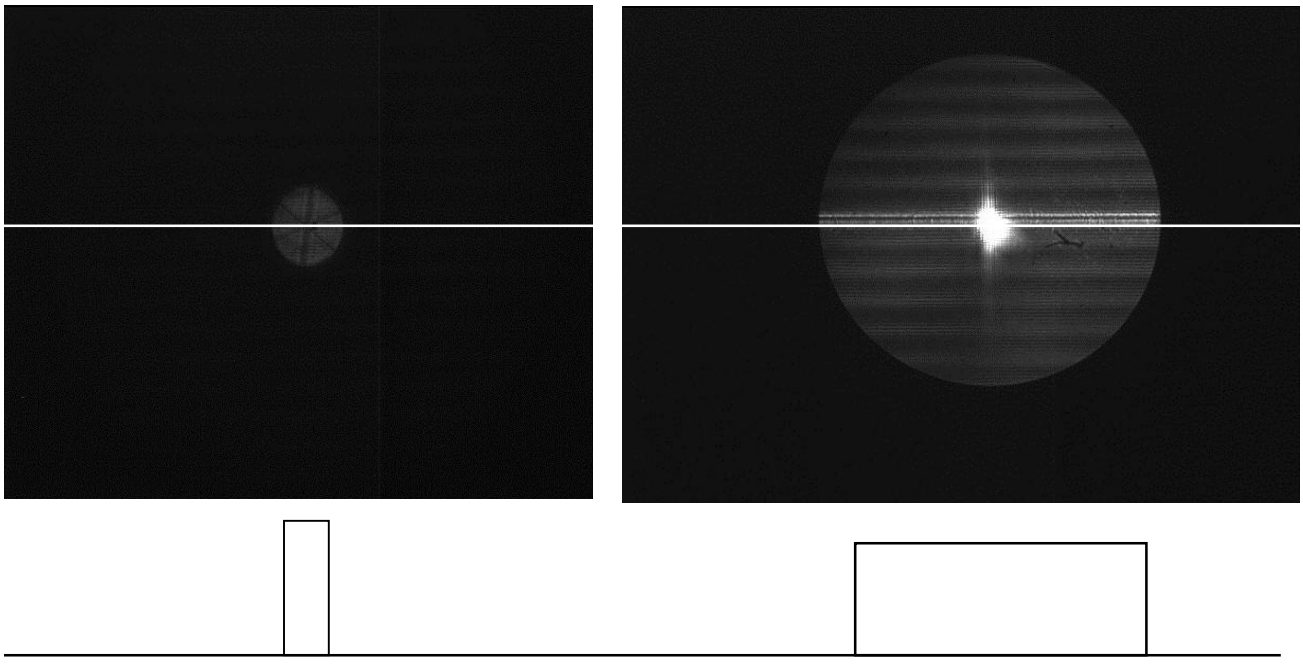

Fig. 6. Cross-section measurement

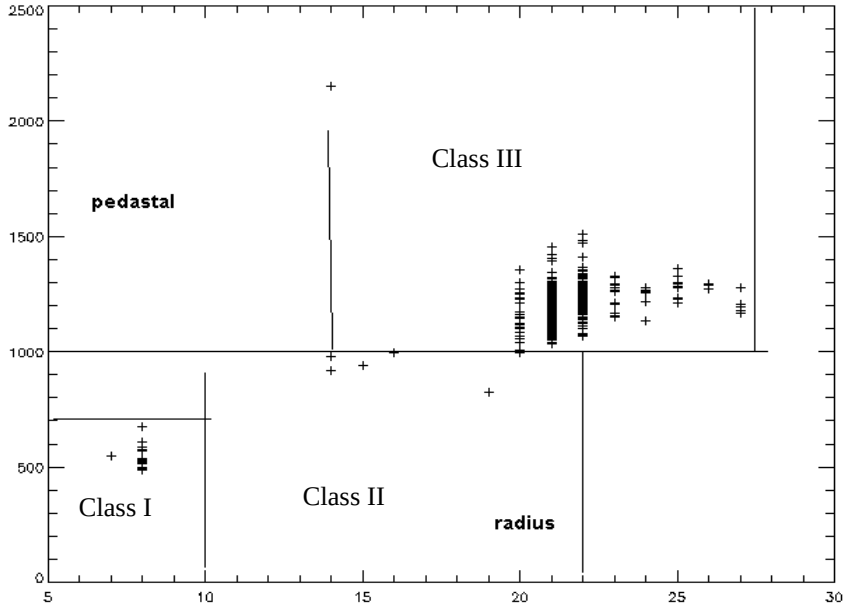

Fig. 7. Pedestal versus chosen radius

Sometimes, the estimation of the radius using a specific characteristic may lead to a wide range of radii. For example, when estimating radius from pedestal measurements such as in the case of defocus [6], the same spot size (or pedestal) may actually represent two different true radii in two instances. Such variations could be due to the variation of the slope of the spot, i.e., top hat versus slowly varying intensity spot. We need to have an overlapping set of values of radii that the algorithm searches through to find the best match. A scatter plot (Fig. 7) of estimated spot size and the final chosen radius explains how and why this is done. Fig. 7 shows the spot size or pedestal along the y-axis and the radius along the x-axis.

The decision for choosing the radii range is based on classifying the estimated spot sizes into three classes. For class I, the spot size is between 500 and 800. The final radius for this class is 7 to 8 pixels. For this range we vary the radius from 5 to 10 pixels. For class II (with spot size between 800 and 1000), we search the range of radii from 11 to 22 pixels. For class III, when the spot size is above 1000 pixels count, the radius could be as low as 14 (for the spot size of 2200 as shown in Fig. 7). Thus for class III, the radius is varied from 14 to 27, which overlaps with some of the class II radii values. For defocused spots, the estimated spot size is a function of the preprocessing used. For example, the window size for the smoothing function and the threshold used for the total pixel calculation, affect the estimated radius calculation and consequently the final result. Any small changes in these steps have to be re-verified with a large set of thousands of real and artificial images to ensure that a pre-processing change does not alter the known acceptable results. Thus, we may have to re-adjust the search space and their boundaries in response to changes in earlier steps.

4. SUMMARY

This paper describes an automated way to determine the most appropriate type and size filter to be used for position detection. Each type is searched sequentially over a search space. Thresholding is used for type discrimination and the maximum of the normalized autocorrelation is used for size determination. Speedup is achieved because the search for the size of the template is done on a segmented image, which contains the type of image being searched. After all the types with their corresponding best sizes are determined, a complete search of the target scene is done for each template. Correlation peak position is used to select the position of the objects.

5. CONCLUSION

Here we describe a generalized method of detecting the position of objects of various shapes and sizes based on a two-or more step approach. In a three-step approach, the range of the search space needs to be determined first. Speedup and methods of reducing position uncertainty are discussed. This method can be extended to any scene containing any number of objects.

ACKNOWLEDGEMENT

The author would like to acknowledge feedback provided by Erlan Bliss on this paper, discussions with Scott Burkhart, Wilbert McClay, and Jim Candy, and technical editing by Sharon Cornelious. This work was performed under the auspices of the U.S. Department of Energy by the University of California, Lawrence Livermore Laboratory under contract No. W-7405-Eng-48.

REFERENCES

1. E. Moses, et al., "The National Ignition Facility: Status and Plans for Laser Fusion and High-Energy-Density Experimental Studies", *Fusion Science and Technology*, Vol. 43, p. 420, May 2003.
2. VanderLugt, "Signal Detection by Complex Spatial Filtering," IEEE Trans. Inf. Theory IT-10, 139-145, 1964.
3. J. L. Horner and J. Leger, "Pattern recognition with binary phase-only filters," *Applied Optics*, Vol. 24, pp. 609-611, 1985.
4. A. A. S. Awwal, M. A. Karim, and S. R. Jahan, "Improved Correlation Discrimination Using an Amplitude-modulated Phase-only Filter," *Applied Optics*, Vol. 29, pp. 233-236, 1990.
5. M. A. Karim and A. A. S. Awwal, Optical Computing: An Introduction, John Wiley, New York, NY, 1992.
6. A. Awwal, J. Candy, C. Haynam, C. Widmayer, E. Bliss, and S. Burkhart "Accurate position sensing of defocused beams using simulated beam templates" In Photonics devices and algorithms for computing VI, Proceedings of SPIE, Vol. 5556, 2004